SANS

A SANS Whitepaper

JumpStart Guide Application Security in Amazon Web Services

Written by **Nathan Getty**

September 2019

Sponsored by:

AWS Marketplace
in conjunction with
Fortinet

Introduction

As organizations begin to transition their applications into cloud environments, security teams must provide application security support and insight during the process.

Today's applications are updated more frequently, and regular release cycles are giving way to more rapid incremental releases. Application development continues to evolve to support a more dynamic release schedule. In response, information security teams must be included in the development process if they are to provide support to development teams. Because organizations plan to deploy applications as soon as they are approved for production, your organization's security team should not be the roadblock.

Because development teams release applications faster than they can be reviewed, it is critical to integrate the skills and guidance of the security team into the development model. Whether the application code is deployed on premises or in a cloud environment, automated security tools provide the information security team with visibility into code as it moves through the developer pipeline. This visibility provides more assurance that security will not be compromised.

This process allows the development teams to remain informed of security concerns for their application as it moves through the pipeline. By embedding security within the build process, your organization can build a strong relationship between the security and development teams. By fostering and developing this relationship, developers and security professionals can work in tandem to deliver secure, timely applications.

According to Forbes, nearly three-quarters of companies are planning to move to a fully software-defined data center within two years. Almost half of businesses are delaying cloud deployment due to a cybersecurity skills gap.[1] This paper seeks to give you a better idea of what your organization needs to successfully plan and execute a secure application transition to, or deployment in, an AWS environment.[2] We discuss how security teams can best support application development teams, what options you have as a security professional for this support, and how best to guide your development teams as they transition workflows to AWS.

Understanding Your Needs

Historically, application development and security teams did not always work closely together. But given the adoption of rapid release cycles and the transition to cloud services, these teams must build a working relationship that effectively supports rapid deployment of secure applications. How can they do that while best using existing tools and processes in the cloud environment?

1. Understand the applications deployed in your organization.

Security analysts need to be knowledgeable about the applications being deployed, at least to the extent of being aware of their primary purpose and target audience. When they understand the application, the underlying code, and for whom the application is designed, they can run threat modeling assessments and plan accordingly. They can make remediation decisions with confidence, bring attention to specific security vulnerabilities, identify which vulnerabilities and risks are acceptable, and provide feedback to the development team. Encouraging security teams to work closely with development teams and speak their language will build a strong, mutually beneficial relationship.

2. Understand application deployment methods within AWS.

Applications can be deployed through any one of several channels or tools. Knowledge of the tools available to development teams can help information security teams define best security practices within those tools and ease incident response or critical changes to the applications. Through awareness of the underlying development process, an organization can be assured that quality information regarding security concerns is being communicated to the development teams.

3. Understand what options and responsibilities you have in AWS as you prepare for securing the application delivery.

The AWS cloud environment gives organizations access to a large developmental toolset in the form of services that include a number of capabilities. Not every service will be a good fit for your organization, so development and security teams should plan ahead and identify which services they will need to use for their application delivery and the security.

---

[1] "2017 State of Cloud Adoption and Security," www.forbes.com/sites/louiscolumbus/2017/04/23/2017-state-of-cloud-adoption-and-security

[2] This paper mentions product names to provide real-life examples of how firewall tools can be used. The use of these examples is not an endorsement of any product.

AWS offers various platforms for setting up such services. For example, AWS offers serverless services, which means your organization is not responsible for operating or maintaining the underlying infrastructure. Although AWS takes full responsibility for operating the hardware, networking and patch management of the underlying infrastructure, responsibility for the security of any application built on the platform lies completely with the organization.

Implementation Options in AWS

AWS offers a number of services and options as well as access to third-party services for secure application development and rapid release cycles.

Cloud-Native Services

When applications and security tools work harmoniously, future problems (and the need to fix them) can be avoided more easily. Fortunately, AWS-native services are built to work well with each other. Leveraging native services can ease the speed of deployment and integration of application security tools. AWS Marketplace contains a collection of ready-to-deploy infrastructure components your organization can deploy directly into their Amazon VPC (Virtual Private Cloud). AWS Marketplace offers a variety of software including, but not limited to, operating systems, network and business intelligence tools, machine learning software, security software and development suites.

The ability to find, test, deploy and validate software through AWS Marketplace helps organizations identify which applications work for them, which allows them to procure and deploy solutions much faster than when having to spend time engaging with a variety of vendors. (Although deploying AWS Marketplace products can be quick and fast, you should still engage with your organization's software onboarding team before deploying new solutions within your environment; your organization may have certain software onboarding procedures even when it comes to native AWS services.) Leveraging native services also has the added benefit of pricing consolidation. Because AWS services are billed to your account with detailed information, organizations can use native services to view all of their AWS costs within a single, detailed page.

Open Source and Custom Solutions

Native services offer direct benefit to your organization, but there may be situations where you prefer custom or open source software (OSS) applications. OSS and custom tools can be leveraged within AWS as long as they are compatible with AWS infrastructure (Microsoft Windows- or Linux-based platforms). For example, it is possible to run custom or OSS solutions on Amazon EC2 (Elastic Cloud Compute). The key difference with EC2 (versus native service) is that your organization inherits the full responsibility for any underlying infrastructure. Your organization is responsible for patch management and any security solutions required for the infrastructure (firewall, intrusion detection and other security tools). Refer to the AWS Shared Responsibility Model[3] for more information.

[3] AWS Shared Responsibility Model, https://aws.amazon.com/compliance/shared-responsibility-model/

Consulting Partner Private Offers

Customers can also engage through Consulting Partner Private Offers (CPPO) to work directly with trusted advisors to select and configure Application Security solutions from AWS Marketplace. As organizations build out their cloud and cloud security strategy and plan, they may want to consider working with partners to accelerate their efforts or fill any gaps in knowledge or resources that are identified. All consulting partners may extend AWS Marketplace third-party solutions directly to customers through CPPO.[4] Not every organization will be able to find resources with deep cloud experience. Even experienced cloud technologists may have experience only with specific industries or cloud vendors. A requirements document could be helpful when approaching prospective consultants.

Needs and Capabilities: The Business Case for Application Security in the Cloud

The benefits of putting applications in the cloud must be balanced by the organization's ability to secure them.

Application Security

The need: Conducting application security assessments and reducing vulnerabilities within the AWS environment

Capabilities

- Increased visibility within the development process and application stack
- Reduced risk and vulnerabilities in the applications before they are deployed
- Automated security assessments with actionable remediation
- A relationship with the development teams

[4] AWS Marketplace Channel Programs, https://aws.amazon.com/marketplace/partners/channel-programs

General AWS Web Application Security Considerations

Regardless of the technology or cloud vendor selected, some general business, technical and operational considerations are associated with implementing application security in the cloud. The following sections highlight many of these considerations.

Business Considerations

	Consideration	Details
	Policies and standards	Organizations must understand their current software development life cycle (SDLC) policy and how it may be affected by a move to a cloud environment. An SDLC policy describes the various stages of application deployment and delivery. These underlying methodologies do not change when moving to a cloud environment but the processes and procedures for application code review, application building, delivery and analysis probably will. Anticipating what changes to the SDLC will be triggered by transitioning to AWS will allow organizations to adopt an SDLC that not only fits the cloud model, but also has tangible benefits for an organization's application delivery within the cloud. Planning and making these changes first will save your organization time should a policy need to be redefined in the future. Organizations should determine the acceptable level of risk for their application(s). Although it would be nice if we could deliver applications without errors or exploitable weaknesses, such a scenario is unfortunately unrealistic. Developers have to release applications within the timelines demanded by their sprints, and they often lack sufficient resources to explore and address all security aspects of their application in the available time. If an organization deploys an application with little or no security validation, it is exposed to a greater risk that the application could be exploited. Organizations must plan ahead and define an acceptable threshold for vulnerabilities within a production-class application. For example: Organization X ships releases for its Acme web app every two weeks. It runs security tests each time the application is built. Its policy states that if those tests find that the application build contains more than three high-risk vulnerabilities or greater than zero critical risk vulnerabilities, Organization X will block application delivery until the issues have been addressed and corrected.
	Licensing options	While AWS operates under the "pay what you use" model, many third-party vendors allow customers to deploy products directly on AWS's infrastructure. Leveraging third-party applications and tools can quickly increase licensing costs for your organization. Take precautions when deploying third-party applications and tools on AWS infrastructure, because your organization will incur both AWS infrastructure usage and software licensing costs. Licensing costs can be charged in a few different ways. They may be billed to the organization on an annual basis or perhaps by the hour. Understanding and planning for expected licensing costs will ensure you are not caught off guard by large invoices from AWS.

Technical Considerations

	Consideration	Details
	Technology deployment	Organizations should plan ways to implement their application security in a repeatable, consumable manner. Security teams can provide guidance in this matter in a variety of ways. Within AWS, applications can be deployed through a fully automated "pipeline"; alternatively, they can be deployed in an ad hoc fashion. An organization would be wise to create small, repeatable security tests as part of the deployment process, and to continuously refine those tests as the application matures. Understanding how your organization deploys its applications will allow the security teams to create and deploy effective security tests that align with the developers' deployment plan. Organizations need to decide if they will allow OSS or unsupported technologies. While it's true that an open source application allows insightful visibility into the application's security, it's also true that open source projects do not come with the luxury of customer support or SLA. If you plan to use open source technology for critical tasks or security assurance, you will need to ensure you have a proper plan in case the tool stops working at some point. On the other hand, OSS tools offer some unique opportunities. Organizations can take advantage of free open source tools and, as their needs outgrow the capabilities, modularity or support level provided by the OSS tooling, they can transition to more professional offerings.

	Consideration	Details
	Application stack	AWS Marketplace offers many tools for securing your organization's applications. Leverage any available open source testing software to get used to integrating security tools into your application development process (and save costs). Static analysis tools (linters) allow you to check your code for programming errors, bugs, stylistic problems and suspicious constructs. Each programming language has its own set of linters, most of which can be installed directly within your developers' preferred integrated development environment (IDE). Having developers use a linter within their IDE saves time in the development process by catching the errors before the application code is pushed. Catching these issues before the application is deployed makes it easier to mitigate them after deployment. Organizations should also consider their application stack and what corresponding Static Analysis Security Testing (SAST) tools might best fit their deployment pipeline. While linters check for bugs, syntactical errors, programmatic errors and code nuances, the purpose of SAST tools is to identify security issues in the application source code (versus during compilation or runtime). As with linters, each language has its own set of SAST tools, so your organization needs to understand the application code being implemented and what the information security teams will need to deploy to validate the codebase.
	Pre-deployment security (inline)	The largest challenge of inline scanning is the time it takes scans to complete. If your organization needs to deploy an application change, your security test should not require a long time to run. Imagine making a small configuration change to your organization's application. You push your code to the development pipeline, and now you have to wait 30 minutes for the security tools to scan your changes. Developers can push these changes many times a day, so waiting for these scans can be frustrating. We recommend that inline scans should not take longer than five minutes (depending on the size of the codebase). Your organization might also want to consider scanning only the changes to the code from the last push (delta scan). This method saves time but may be better suited to more mature organizations. It also makes sense to occasionally scan the entire codebase outside of the pipeline (out-of-band scans). We advise that organizations take small, repeatable, incremental steps in deploying inline scans for application pipelines. It's a good idea for your security team to have its own source code repository where it stores its tests. After a test has been created and validated, it can be stored in the repository. Once the code is in this repository, it may be shared with the developers, and they can include them within their development pipeline. You can work with the developers to ensure that the latest copy of the security test is always referred to when inline scanning. This procedure allows the security team to update the test as it sees fit. Because the development team has the latest copy of the test always being pulled into the pipeline, there should be no additional work when the security tests are updated. Leveraging this approach allows you to continuously test applications, update the tests and keep track of what exactly was changed via revision control.
	Post-deployment security (out of band)	Organizations will need to decide when to implement post-deployment security scanning. We mentioned out-of-band scanning earlier: If scans take too long to complete, they can be scheduled after the application has been deployed. Full scans by Dynamic Application Security Testing (DAST) tools can take hours to run, depending on the application size and scope of the scan. The following are examples of tools that should be run outside of the deployment pipeline: • **Infrastructure scans—**These can take a long time depending on the scope of the resources and security checks the scan performs. • **Dynamic application security scans—**These require the environment to already be up and running. Like infrastructure scans, these scans can take some time to complete, depending on the organization's scanning scope. • **Full web application security scans—**Depending on the parameters of the test (credentialed/no-credential/spider/full active scan) and the size of the application, this scan can take a long time to run and should not be used inline. Organizations will need to decide what is necessary to test and ensure application security for applications that have already been deployed. Solutions such as infrastructure security scanning, WAF implementation and DDoS protection should be evaluated.

Operational Considerations

	Consideration	Details
	Processes and procedures	Organizations may need to create or modify processes and procedures for security web applications in AWS. While some existing processes and procedures may work without modification, hosting applications in AWS means different methods of application delivery. Organizations may want to start to include developers and key individuals involved with application delivery in meetings and discussions about application security testing. Security teams might also want to sit in on development meetings and inform discussions when application security concerns arise.
	Resources and deployment synergy	Security in AWS and the applications deployed within the cloud will take dedicated resources to ensure that the proper policies and procedures are followed. Organizations must be cognizant that resources will need to be dedicated in such an effort, and they should plan accordingly. Organizations should consider which approach they would like to take with their cloud application security and the level of responsibility for each team involved within the process. Development and security teams within your organization need to take responsibility for the security and integrity of the application.

AWS Implementation Considerations

Application Security

	Consideration	Details
	Cloud context support	Application deployment leverages many ephemeral resources that support application delivery. Catalog all possible resources used within the deployment process for identifying any issues. **Evaluate:** • The additional cloud context (tags or image IDs, other possible ephemeral resources) captured within the development processes (phoenix servers, artifacts and the like) • Logging and cataloging of the cloud resources for traceability and troubleshooting
	Deployment	Deployment methods for security tools within AWS can vary depending on the development pipeline. Organizations should deploy these tools within the context of the development pipeline. **Evaluate:** • Installation and initial configuration for tools • Possible use of professional services to aid or accelerate tool deployment • Programming tools and languages used in the applications and their corresponding DAST/SAST tools • The availability of managed or SaaS components or preconfigured appliances from AWS Marketplace • Leveraging AWS-native services for security implementation
	Integration	Integration of application security tools into current processes/procedures ensures security teams can respond to risks. Integrating application security tools into the development pipeline allows for visibility, deployment and management. It also provides ease of use for security and development teams. **Evaluate:** • The development pipeline process and how to embed security tools and scans inline within a reasonable time • Tools that integrate with current security solutions (SIEM, SOAR, IT service management) • API support (REST APIs available, SOAP APIs available, other available programmatic APIs) • Use of custom plugins or integrations • Integrations with native AWS services

Making the Choice

To summarize, the key considerations for implementing application security in AWS are:

- Cloud context
- Deployment
- Integration
- Configuration and iteration
- Reporting

Evaluate Your Organization's Current Deployment Process

There are many ways to deploy applications with AWS, and many methods with which to build out your deployment pipeline. When defining your proof of concept, include significant members of the application deployment team and ensure you understand their method of deployment infrastructure (Amazon EC2, Amazon ECS, serverless) and deployment pipelines (AWS CodePipeline, Jenkins, other deployment tools). Once your organization has a strong understanding of the deployment process, it can begin to evaluate its needs and considerations for security tools. Define a proof of concept that meets both your organization's considerations and the developers' current deployment process.

Define a Plan and Implement

By defining and understanding its cloud architecture, risk profile, business requirements and available resources as well as all the possible deployment methods within AWS, an organization should have a clear idea of its road map for application security protection. Understand that defining application security that meets all the discussed considerations is nearly impossible, so define and use what works best for your organization.

The best course of action is to define a proof-of-concept plan based on the considerations and implementation options. Ensure that your organization's development team is included in this process, because they will have a very strong understanding of the application and which security concerns to note. Once you have planned, developed and validated your POC, development and security teams can start defining a repeatable process for integrating app security within the development process. In this stage, your organization should work with the development team to identify the team's current security issues and how the developed POC will help secure the application and reduce the application's risk to meet the organization's risk threshold.

Conclusion

Application security is a crucial step for organizations' cloud security strategy. Having a defined plan and integrating security within the development process allows for greater visibility within the application delivery process, visibility into the security stance of the application and a defined remediation process for application security vulnerabilities.

Work with the development team through each stage of the DevOps life cycle (see Figure 1). Plan with the developers, join meetings when they develop and discuss their applications, ask the developer team for help when writing security tests in the verification stage, add out-of-band security (WAF protections, EDR solutions, DDoS protections and the like) in the release stage, and constantly monitor the security state of the application through your infrastructure monitoring and log analysis. Security tools and checks can be applied to many stages of the development process.

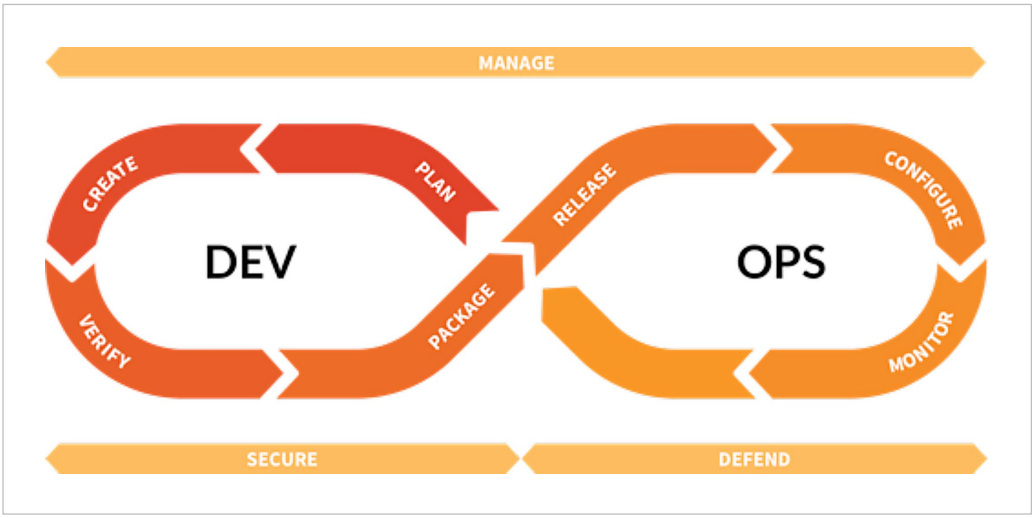

Figure 1. The DevOps Life Cycle

Keep in mind that this process should always be repeatable and easy to use. Start small and build from there. To get started today, consider an evaluation of some of the solutions readily available via the AWS Marketplace. You may also consider leveraging a SaaS solution to jump-start your organization's journey into AWS application security.

About the Author

Nathan Getty holds the GWAPT and GCIA certifications, and he recently won the SANS Cyber Defense NetWars competition, a defense-focused challenge that tests the ability to solve problems and secure systems from compromise. Nathan currently works in the Canadian insurance industry. In his organization, he focuses on cloud security, including AWS onboarding, and developing best security practices and general security/cloud insights. Nathan also focuses on driving DevOps methodologies in the company's security program, implementing continuous delivery platforms to allow smoother development and improvement of internal security applications.

Sponsor

SANS would like to thank this paper's sponsor:

aws marketplace

in conjunction with

FORTINET®

About Fortinet

Fortinet breaks down the barriers that inhibit security visibility and management across private, public, and hybrid cloud platforms. The Fortinet Security Fabric for AWS helps organizations maintain consistent security protection in a shared responsibility model, from on-premises to the cloud. It delivers comprehensive and fully programmable multilayered security and threat prevention capabilities for AWS users. At the same time, it streamlines operations, policy management, and visibility for improved security lifecycle management with full automation capabilities. Visit **www.fortinet.com/aws** for information on Fortinet's application security solutions.